Summary

We want to create a recipe creating/sharing and grocery list app. You’ll be planning out what tables we’ll need, what information they’ll store, and how the data will relate to each other.

Features

* users can sign into the app with their email and password
* users can create recipes with ingredients and instructions
* recipes can be marked as public or private
* users can view other people’s recipes
* ingredients from recipes can be added to user’s grocery lists
* users can create their own occasions and assign recipes to occasions

## Brainstorming

Sign in Process

* Email
* Password
* Username
* First name
* Last name
* Birthday
* Location
* Interests
* Profile picture
* User ID
* Allergies

Create recipes

* Recipe ID
* User ID (link to author)
* Views count
* Average rating
* Ingredients
* Instructions
* Public/private visibility
* Allergens
* Difficulty level
* Time required
* Servings made

Favorites

* Recipes ID
* User ID

Review recipes/Comments

* Notes
* Attempts
* Modifications

Share recipes and view others

* User ID 1
* User ID 2
* Recipes ID
* Comments
* FriendList ID

Ingredients and instructions

* Ingredients list
* Tools, equipment

User-created occasions

* User ID
* Occasion ID
* Occasion name

Assign recipes to occasions

Grocery Lists

* User ID
* List ID
* Ingredients

## Table Ideas

Users: this table will hold information about the users, each row will be a unique individual user

User Details: this table elaborates on more details on the profile of each user that differentiates from bare-bones login credentials

Recipes: this table will hold information about recipes, each row will be a unique recipe

Reviews: this table will hold information on comments, ratings, and reviews of recipes with each row being a unique review. Reviews would feed into recipes.

Shared List/Friends: users can share recipes between each other. This will work like a friends list, maybe an association table

Favorites: users can designate a subset of their recipes as favorites. Each row will be a different recipe under the same user.

Ingredients: a table that has a library of ingredients found in the recipes. This will feed into the recipes. Recipes can overlap ingredients. Recipes can have many ingredients.

Occasions: a table that will hold information about various occasions that users will specify. Associate with recipes and users.

User-Occasion and/or Recipe-Occasion? Users will contribute to occasions and populate the Occasions Table. Occasions can be associated with specific recipes in another association table?

Grocery Lists: a table that will hold grocery list entries generated by users.

Grocery Lists\_Content: a table that will hold the contents of each grocery list.

## Relationships

### One-to-one

Users to User Details because each user only has one set of details.

### One-to-many

Users to Recipes because one user can create multiple recipes, but each recipe is only authored by one user.

Recipes to Reviews because one recipe can have many reviews but each review can only address one recipe.

Users to Reviews because users can leave many reviews, but each review can only have one user to compose it.

Users to Grocery Lists because one user can create multiple grocery lists in their account but each grocery list is generated by one person.

Grocery List to Grocery List\_Contents because each grocery list will only have one bin of contents within but

### Many-to-many

Recipes to Ingredients because recipes can contain many ingredients and ingredients can be used in many recipes.

Friends List/Shared Recipes because users can have many friends and each friend can also be friends with many users.

Users to Favorites because one user can list multiple favorite recipes and each recipe can be favorited by multiple users.

Users to Occasions because one user can create multiple occasions and each occasion can be referenced by multiple users.

Occasions to Recipes because occasions can be linked to multiple recipes and many recipes can be associated with every occasion.

Grocery Lists\_contents to Ingredients because each grocery list can contain multiple ingredients and ingredients can be included in many different grocery lists.

## Columns

##### Users

* User-id:
* Email:
* Password:
* username:

##### UserDetails

* User-id:
* First-name:
* Last-name:
* Birthday:
* Location:
* Interests:
* Profile-picture:
* allergies:

##### Recipes

* Recipe-id:
* User-id:
* Ingredients:
* Instructions:
* Public:
* Views:
* Average-rating
* Difficulty:
* Time-required:
* Servings:

##### Occasions

* Occasion-id:
* User-id:
* Occasion-name:

##### OccasionsRecipes

* Occasion-recipe-id:
* Recipe-id:
* occasion-id:

##### GroceryList

* Grocery-list-id:
* User-id:
* ingredients:

##### Ingredients

* Ingredient-id:
* ingredient-name:

##### Reviews

* Review-id:
* Recipe-id:
* User-id:
* Body:
* rating:

##### FriendsRecipes

##### FavoriteRecipes

CREATE TABLE "public.Users" (

"user\_id" serial NOT NULL,

"email" varchar(100) NOT NULL,

"password" varchar(32) NOT NULL,

"username" varchar(30) NOT NULL UNIQUE,

CONSTRAINT "Users\_pk" PRIMARY KEY ("user\_id")

) WITH (

OIDS=FALSE

);

CREATE TABLE "public.UserDetails" (

"user\_detail\_id" serial NOT NULL,

"user\_id" integer NOT NULL,

"first\_name" varchar(100) NOT NULL,

"last\_name" varchar(100) NOT NULL,

"birthday" DATE,

"location" varchar(50),

"interests" TEXT,

"profile\_picture" varchar,

"allergies" TEXT,

CONSTRAINT "UserDetails\_pk" PRIMARY KEY ("user\_detail\_id")

) WITH (

OIDS=FALSE

);

CREATE TABLE "public.Recipes" (

"recipe\_id" serial NOT NULL,

"user\_id" integer NOT NULL,

"instructions" TEXT NOT NULL,

"public" BOOLEAN NOT NULL,

"views" integer NOT NULL,

"average\_rating" DECIMAL NOT NULL,

"difficulty" integer NOT NULL,

"time\_required" integer NOT NULL,

"servings" integer NOT NULL,

"allergens" varchar NOT NULL,

CONSTRAINT "Recipes\_pk" PRIMARY KEY ("recipe\_id")

) WITH (

OIDS=FALSE

);

CREATE TABLE "public.Occasions" (

"occasion\_id" serial NOT NULL,

"user\_id" integer NOT NULL,

"occasion\_name" varchar(50) NOT NULL,

CONSTRAINT "Occasions\_pk" PRIMARY KEY ("occasion\_id")

) WITH (

OIDS=FALSE

);

CREATE TABLE "public.OccasionsRecipes" (

"occasion\_recipe\_id" serial NOT NULL,

"recipe\_id" integer NOT NULL,

"occasion\_id" integer NOT NULL,

CONSTRAINT "OccasionsRecipes\_pk" PRIMARY KEY ("occasion\_recipe\_id")

) WITH (

OIDS=FALSE

);

CREATE TABLE "public.Ingredients" (

"ingredient\_id" serial NOT NULL,

"ingredient\_name" varchar(50) NOT NULL,

"ingredient\_price" integer NOT NULL,

"description" TEXT NOT NULL,

CONSTRAINT "Ingredients\_pk" PRIMARY KEY ("ingredient\_id")

) WITH (

OIDS=FALSE

);

CREATE TABLE "public.GroceryList" (

"grocery\_list\_id" serial NOT NULL,

"user\_id" integer NOT NULL,

"ingredient\_id" integer NOT NULL,

CONSTRAINT "GroceryList\_pk" PRIMARY KEY ("grocery\_list\_id")

) WITH (

OIDS=FALSE

);

CREATE TABLE "public.FriendsRecipes" (

"friendlist\_id" serial NOT NULL,

"user\_id\_1" integer NOT NULL,

"user\_id\_2" integer NOT NULL,

"recipe\_id" integer NOT NULL,

CONSTRAINT "FriendsRecipes\_pk" PRIMARY KEY ("friendlist\_id")

) WITH (

OIDS=FALSE

);

CREATE TABLE "public.Reviews" (

"review\_id" serial NOT NULL,

"recipe\_id" integer NOT NULL,

"user\_id" integer NOT NULL,

"body" TEXT NOT NULL,

"rating" integer NOT NULL,

CONSTRAINT "Reviews\_pk" PRIMARY KEY ("review\_id")

) WITH (

OIDS=FALSE

);

CREATE TABLE "public.FavoriteRecipes" (

"favorite\_id" serial NOT NULL,

"user\_id" integer NOT NULL,

"recipe\_id" integer NOT NULL,

CONSTRAINT "FavoriteRecipes\_pk" PRIMARY KEY ("favorite\_id")

) WITH (

OIDS=FALSE

);

CREATE TABLE "public.RecipeIngredients" (

"recipe\_ingredient\_id" serial NOT NULL,

"recipe\_id" integer NOT NULL,

"ingredient\_id" integer NOT NULL,

CONSTRAINT "RecipeIngredients\_pk" PRIMARY KEY ("recipe\_ingredient\_id")

) WITH (

OIDS=FALSE

);

ALTER TABLE "UserDetails" ADD CONSTRAINT "UserDetails\_fk0" FOREIGN KEY ("user\_id") REFERENCES "Users"("user\_id");

ALTER TABLE "Recipes" ADD CONSTRAINT "Recipes\_fk0" FOREIGN KEY ("user\_id") REFERENCES "Users"("user\_id");

ALTER TABLE "Occasions" ADD CONSTRAINT "Occasions\_fk0" FOREIGN KEY ("user\_id") REFERENCES "Users"("user\_id");

ALTER TABLE "OccasionsRecipes" ADD CONSTRAINT "OccasionsRecipes\_fk0" FOREIGN KEY ("recipe\_id") REFERENCES "Recipes"("recipe\_id");

ALTER TABLE "OccasionsRecipes" ADD CONSTRAINT "OccasionsRecipes\_fk1" FOREIGN KEY ("occasion\_id") REFERENCES "Occasions"("occasion\_id");

ALTER TABLE "GroceryList" ADD CONSTRAINT "GroceryList\_fk0" FOREIGN KEY ("user\_id") REFERENCES "Users"("user\_id");

ALTER TABLE "GroceryList" ADD CONSTRAINT "GroceryList\_fk1" FOREIGN KEY ("ingredient\_id") REFERENCES "Ingredients"("ingredient\_id");

ALTER TABLE "FriendsRecipes" ADD CONSTRAINT "FriendsRecipes\_fk0" FOREIGN KEY ("user\_id\_1") REFERENCES "Users"("user\_id");

ALTER TABLE "FriendsRecipes" ADD CONSTRAINT "FriendsRecipes\_fk1" FOREIGN KEY ("user\_id\_2") REFERENCES "Users"("user\_id");

ALTER TABLE "FriendsRecipes" ADD CONSTRAINT "FriendsRecipes\_fk2" FOREIGN KEY ("recipe\_id") REFERENCES "Recipes"("recipe\_id");

ALTER TABLE "Reviews" ADD CONSTRAINT "Reviews\_fk0" FOREIGN KEY ("recipe\_id") REFERENCES "Recipes"("recipe\_id");

ALTER TABLE "Reviews" ADD CONSTRAINT "Reviews\_fk1" FOREIGN KEY ("user\_id") REFERENCES "Users"("user\_id");

ALTER TABLE "FavoriteRecipes" ADD CONSTRAINT "FavoriteRecipes\_fk0" FOREIGN KEY ("user\_id") REFERENCES "Users"("user\_id");

ALTER TABLE "FavoriteRecipes" ADD CONSTRAINT "FavoriteRecipes\_fk1" FOREIGN KEY ("recipe\_id") REFERENCES "Recipes"("recipe\_id");

ALTER TABLE "RecipeIngredients" ADD CONSTRAINT "RecipeIngredients\_fk0" FOREIGN KEY ("recipe\_id") REFERENCES "Recipes"("recipe\_id");

ALTER TABLE "RecipeIngredients" ADD CONSTRAINT "RecipeIngredients\_fk1" FOREIGN KEY ("ingredient\_id") REFERENCES "Ingredients"("ingredient\_id");